

Tabla de Abstracción – Proyecto “EPS Salvando Vidas”

Nombre del Estudiante: Sergio Efren Bolaños Acosta

Planteamiento del Problema:

La EPS “Salvando Vidas” requiere una aplicación de escritorio que permita registrar y controlar los datos de sus usuarios, diferenciando el tipo de atención (medicina general o examen de laboratorio) y calculando automáticamente el valor del copago según el estrato socioeconómico.

La aplicación debe permitir registrar la información de los usuarios en tres estructuras de datos diferentes (pila, cola y lista), visualizar los registros en una tabla (Treeview), generar reportes específicos y eliminar registros conforme al comportamiento de cada estructura.

Nombre de la clase y su ámbito de visibilidad (public o private)	Nombre de las propiedades y/o atributos con sus tipos de datos	Tipo de estructura que soporta (Pila, Cola o Lista)	Acciones que tendrá el formulario de ingreso de datos (métodos)
Public class EstructuraDatosUsuario	tipo_id : str numero_id : str nombre : str edad : int estrato : int atencion : str copago : int fecha : str	Soporta las tres estructuras: Pila (list) Cola (collections.deque) Lista (list)	Métodos de la clase: • as_tuple() → Retorna los datos del usuario en formato de tupla para mostrarlos en la tabla. Métodos de la aplicación (EPSApp): • check_password() → Valida el acceso mediante contraseña. • acerca_de() → Muestra información del curso y estudiante. • open_main_window() → Abre el formulario principal tras el login. • registrar() → Registra un nuevo usuario en la estructura seleccionada (apilar, encolar o agregar). • limpiar_campos() → Limpia los campos del formulario sin borrar los registros existentes. • recalcular_copago() → Calcula automáticamente el copago según el estrato y tipo de atención. • validar_campos() → Verifica que los datos ingresados sean válidos (numéricos o alfabéticos según el campo). • mostrar_reporte() → Muestra reportes dependiendo de la

			<p>estructura:</p> <ul style="list-style-type: none"> - Pila → suma de copagos - Cola → cantidad de registros - Lista → promedio de edades <p>• eliminar_registro() → Elimina un registro según el comportamiento de la estructura:</p> <ul style="list-style-type: none"> - Pila → desapilar - Cola → desencolar - Lista → eliminar usuario buscado <p>• confirmar_salida() → Solicita confirmación antes de cerrar la aplicación.</p>
--	--	--	---

Clase principal: EstructuraDatosUsuario encapsula los datos individuales de cada usuario registrado.

Estructuras utilizadas:

list → implementa la **pila (LIFO)** y la **lista**.

collections.deque → implementa la **cola (FIFO)**.

Interfaz gráfica: Desarrollada con tkinter, gestiona las acciones de registro, eliminación, cálculo y reporte.

Relación entre clases:

La clase EPSApp contiene instancias de EstructuraDatosUsuario que son almacenadas en las estructuras de datos y mostradas en la interfaz gráfica mediante Treeview.